Program File:

```
//These lines are defining symbolic names for various GPIO pins on the ESP32 board. These names
//are used to make the code more readable and maintainable.

#define D_23_SDA 23

#define D_22_SCL 22

#define D_25_A28 25

#define D_26_A29 26

#define D_27_A27 27

#define TX_0 01

#define RX_0 03

#define D_21 21

#define D_19 19

#define D_18 18

#define D_05 05

#define TX_2 17

#define RX_2 16

#define D_04_A20 04

#define D_02_A22 02

#define D_15_A23 15

#define D_12_A25 12

#define D_13_A24 13

#define D_14_A26 14

#define V_P_A10 36

#define V_N_A13 39

#define D_32_A14 32

#define D_33_A15 33

#define D_34_A16 34

#define D_35_A17 35
```

//These lines define Blynk configuration settings such as the template ID, template name, and authentication token. It also includes necessary libraries for WiFi communication and Blynk integration.//

```
#define BLYNK_TEMPLATE_ID "TMPL3Qoacc5-c"

#define BLYNK_TEMPLATE_NAME "iot based wireless vehicle charging station"

#define BLYNK_AUTH_TOKEN "oOMXRVMxN-dwV3ajowRUfkqk1hZtmGdo"

#define BLYNK_PRINT Serial


#include <WiFi.h>

#include <WiFiClient.h>

//#include <ESP32WiFi.h>

#include <BlynkSimpleEsp32.h>
```

// Variables for Blynk authentication and WiFi credentials are declared.

```
char auth[] = BLYNK_AUTH_TOKEN;

char ssid[] = "BestProject";

char pass[] = "12345678";
```

//Variables sense_2 and sense_4 are defined to represent the GPIO pins connected to sensors for detecting vehicles at two different spots.

```
int sense_2 = 15, Ir;

int sense_4 = 4, Ir2;

/*


  #include <Wire.h>


  #include <LiquidCrystal_I2C.h>

  LiquidCrystal_I2C lcd(0x27, 16, 2); //sometimes the LCD adress is not 0x3f. Change to 0x27 if it dosn't work.

*/
```

//This line includes the LiquidCrystal library for controlling LCD displays and initializes an instance of the LiquidCrystal class for the LCD display connected to specific GPIO pins.//

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(D_05, D_18, D_19, D_21,  D_22_SCL,  D_23_SDA);

//              ( RS,    E ,   D4,   D5,       D6,       D7)

int p = 0;

int val;
```

//These lines define symbolic names for PWM pins used to control the charging mechanisms for the two spots.//

```
#define PWM_12 12

#define PWM_13 13

int brightness = 0;    // how bright the PWM_12 is

int fadeAmount = 10;    // how many points to fade the PWM_12 by

int frequency = 10000;

bool charge_flag_1 = 0, charge_flag_2 = 0;




WidgetLED S_S1(V3);  WidgetLED S_S2(V4);

BLYNK_WRITE(V1)


{


  int v1 = param.asInt();



  if (v1 == 1)  {
```

//Flags to indicate whether charging is requested for spot 1 and spot 2 are initialized.

```
    charge_flag_1 = 1;
```

```
  }

  else  {

    charge_flag_1 = 0;

  }

}
BLYNK_WRITE(V2)
{

  int v2 = param.asInt();
  if (v2 == 1)  {

    charge_flag_2 = 1;

  }

  else  {

    charge_flag_2 = 0;

  }

}
void setup()
{

  pinMode(PWM_12, OUTPUT);
  pinMode(PWM_13, OUTPUT);
  //  lcd.init();                //Init the LCD

  //  lcd.backlight();           //Activate backlight



  lcd.begin(16, 2);



  // Serial.begin(9600);



  pinMode(sense_2, INPUT);

  pinMode(sense_4, INPUT);



  //  pinMode(D_26_A29, INPUT);
  Blynk.begin(auth, ssid, pass);
#define scrollength 40
  for (int i = scrollength; i > 15; i--)

  {
    lcd.clear();
    delay(70);
    lcd.setCursor(i, 1);   lcd.print("      wireless vehicle charging station");
    delay(50);
  }
}
```

```
void loop()

{
  lcd.setCursor(0, 0);   lcd.print("Spot1      Spot2");

  ////////////////////  Spot 1    ////////////////////

  if (digitalRead(sense_2) == LOW)

  {

    S_S1.on();



    if (charge_flag_2 == 1)

    {

      lcd.setCursor(0, 1);   lcd.print("Charge");

      analogWrite(PWM_13, 110);

    }

    else

    {

      lcd.setCursor(0, 1);   lcd.print(" OFF  ");

      analogWrite(PWM_13, 0);
    }
  }
  else
  {
    lcd.setCursor(0, 1);    lcd.print(" OFF  ");
    analogWrite(PWM_13, 0);

    S_S1.off();
  }
  ////////////////////  Spot 2   /////////////////////

  if (digitalRead(sense_4) == LOW)

  {
    S_S2.on();

    if (charge_flag_1 == 1 )
    {

      lcd.setCursor(10, 1);   lcd.print("Charge");

      analogWrite(PWM_12, 110);

    }

    else

    {

      lcd.setCursor(10, 1);    lcd.print(" OFF  ");

      analogWrite(PWM_12, 0);

    }


  }

  else

  {

    lcd.setCursor(10, 1);

    lcd.print(" OFF  ");

    analogWrite(PWM_12, 0);
```

```
    S_S2.off();


  }
}
```